Claim Amendment

Please amend the claims as follows (marked version):

1. (currently amended) A network interface, comprising:

a direct memory access unit; and

circuitry to:

receive and transmit network data;

maintain a set of statistics metering operation of the network interface, the set of statistics including at least one selected from the group of: (1) a number of bytes received, and (2) a number of packets received;

~~receive a periodic time value to perform periodic direct memory access transfers of the maintained set of statistics to a host processor memory; and~~

periodically initiate [[a]] direct memory access transfers of the set of statistics to the host processor memory at a periodicity of ~~the periodic~~ a time interval value; and

configure said initiation of the direct memory access transfers using a configuration information, by intercepting one or more packets received from said host traveling along a transmit path and determining said configuration information from a payload of said one or more packets, wherein said configuration information comprises said time interval value.

2. (previously presented) The network interface of claim 1, wherein the set of statistics comprises each of the following: a number of packets received by the interface,

a number of bytes received by the interface, a number of packets transmitted by the interface, and a number of bytes transmitted by the interface.

3. (previously presented) The network interface of claim 2, wherein the circuitry comprises circuitry to include a timestamp with the direct memory access transfer of the set of statistics, the timestamp being a time when values of the set of statistics transferred by direct memory access were set by the network interface.

4. (previously presented) The network interface of claim 2, wherein the circuitry comprises circuitry to include a sequence count with the direct memory access transfers of the at least one statistic, the sequence count sequentially numbering successively DMA-ed sets of the statistics.

5. (previously presented) The network interface of claim 1, wherein the set of statistics comprises multiple RMON (Remote Monitoring) statistics.

6. (previously presented) The network interface of claim 1, wherein the circuitry comprises circuitry to initiate direct memory access transfer of received network data.

7. (original) The network interface of claim 1, wherein the network interface comprises a framer.

8. (original) The network interface of claim 7, wherein the network interface comprises a Media Access Controller (MAC).

9. (original) The network interface of claim 1, wherein the network interface comprises a PHY.

10. (previously presented) The network interface of claim 1, further comprising circuitry to configure the circuitry to initiate the direct memory access transfers.

11. (cancelled)

12. (original) The network interface of claim 10, wherein the circuitry to configure comprises at least one register.

13. (currently amended) The network interface of claim 10, wherein the circuitry to configure comprises circuitry to determine said configuration information from said one or more received packets.

14. (currently amended) The network interface of claim 13, wherein the circuitry to determine said configuration information from said one or more received packets comprises circuitry to intercept packets received from the host traveling along [[a]] said transmit path.

15. (original) The network interface of claim 1, wherein the direct memory access unit comprises circuitry to notify a processor of completion of a transfer.

Claims 16-38. (canceled)

Remarks

Per the discussions at the personal interview with the examiners, on March 5, 2009, claim 1 is further amended to emphasize (a) DMA transfer of the statistics from NIC card to memory, (b) periodic push by NIC card without requiring a query from CPU for every time statistics is pushed, and (c) getting the configuration information for the DMA periodic push from the payload of the intercepted packet(s).

The examiner's cited prior art, US 6,434,620 (Boucher et al), teaches direct memory access (DMA) transfer of received network data into the main memory (Col. 63, lines 16-43); however, *Boucher* does not teach the DMA transfer of the statistics. Instead, *Boucher* teaches querying the INIC by CPU (Col. 44, lines 20-25) (Note that Microsoft Driver is running on the host side, i.e., the host's CPU). In fact, *Boucher* teaches inefficiencies of using netstat invocation (Col. 44, lines 27-30) which may result in thousands of repeated queries. However, the solution proposed by *Boucher* significantly differs from our invention. *Boucher* still repeats queries after every time interval (Col. 44, by noting the timestamp of the previous query to the INIC. So, for example, if a user/application requires the statistics every 100 msec, *Boucher*'s solution would require tens of thousands of queries to INIC by the host's CPU in an hour. On the other hand, in our invention, once the host sends the interval information to NIC (via packets), the periodic DMA push of statistics does not require any repeated query from the CPU.

We disagree with the examiner's official notice that "receive a periodic time value to perform periodic direct memory access transfers of the maintained set of statistics to a host processor memory;" was well-known. As explained in previous paragraph, *Boucher*, for example, proposed a significantly different way to periodically obtain the statistics, which amounted to having the periodic time value maintained and used on the host side when repeatedly querying INIC, as opposed to receiving the time value at the NIC card. In amended claim 1, this language is eliminated to streamline the claim language;

however, the essence of the limitation is still maintained in the claim language when receiving the configuration information from the intercepted packet.

We disagree with the examiner's official notice that "periodically initiate a direct memory access transfer of the set of statistics to the host processor memory at a periodicity of the periodic time value" was well-known. As explained in previous paragraphs, *Boucher*, for example, proposed a significantly different way to periodically obtain the statistics, which amounted to repeatedly querying INIC from the host, as opposed to having INIC push such statistics without requiring repeated query from host/CPU. Not only *Boucher* does not teach DMA of the statistics from the NIC into the host's memory (as explained earlier), it also does not teach periodic initiation of such action by the NIC. Thus, on both features above, we respectfully challenge the official notice taken by the examiner, especially in view of the current amendment and the clarifications at the interview.

Boucher does not teach the configuration of INIC card by the host via a payload of an intercepted packet (by INIC). *Boucher* teaches (Col. 22, lines 3-10) INIC sending a "NULL" interlock frame to host's CPU in order to signal the end of slow-path frames. Therefore, (1) such a packet is not sent from the host to INIC, in contrast to our invention, (2) such packet does not contain the configuration information for INIC (for DMA initiation), in contrast to our invention, and (3) such packet is not intercepted in INIC, but received by the host (drivers), in contrast to our invention, where the configuration packet is intercepted by NIC, and not the host.

Boucher does not teach the intercepting configuration packet in NIC. *Boucher* teaches a filter driver (Fig, 7, item 210; Col. 35, line 66 through Col. 36, line 20), which attaches (or hooks) to other drivers for processing requests. All such drivers in *Boucher* are running on the host side, i.e., utilizing the host's CPU. Therefore, such teaching is not applicable to our invention, where the NIC (not the host) intercepts the packet. In fact, this teaches away from our invention, because if the host, instead of the NIC, filters the packet for the configuration information, the host/CPU still needs to communicate that

information to the NIC, which was the point of intercepting the packet(s) by the NIC in our invention, in the first place.

Claim 1 was amended to streamline the claim language (with regards to receiving the configuration information including the time interval value), add more structure and tie the configuration information to the time interval value, as discussed during the personal interview.

The other prior art of record do not teach the features mentioned above, either.

Therefore, we believe that claim 1, as amended, is patentable over the cited prior art of record.

The dependent claims 2-10, 12-15 (with parent claim 1) are therefore patentable over cited prior art, as well.

Claims 13 and 14 are amended to correct for antecedent basis with respect to the currently amended claim 1.

Claims 16-38 are canceled.

Thus, all pending claims are allowable, and we respectfully request an allowance of this application.